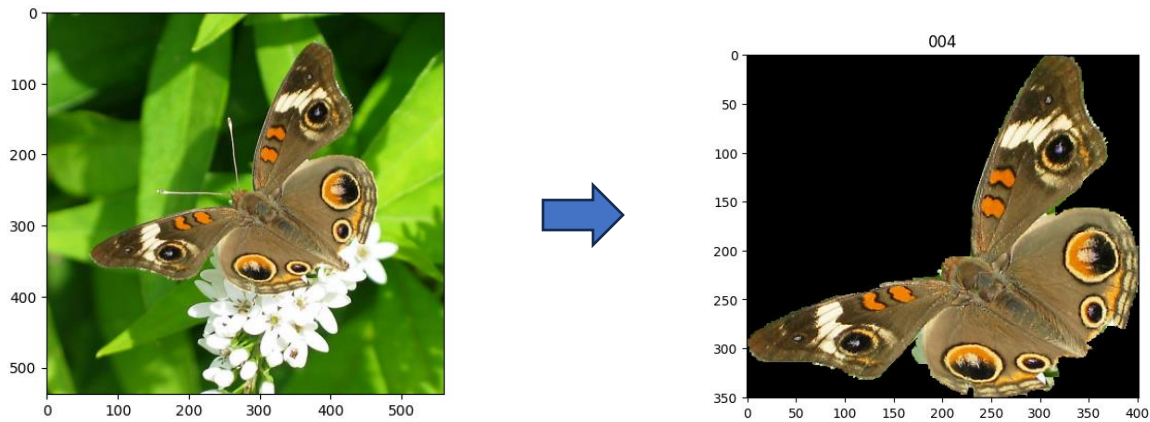
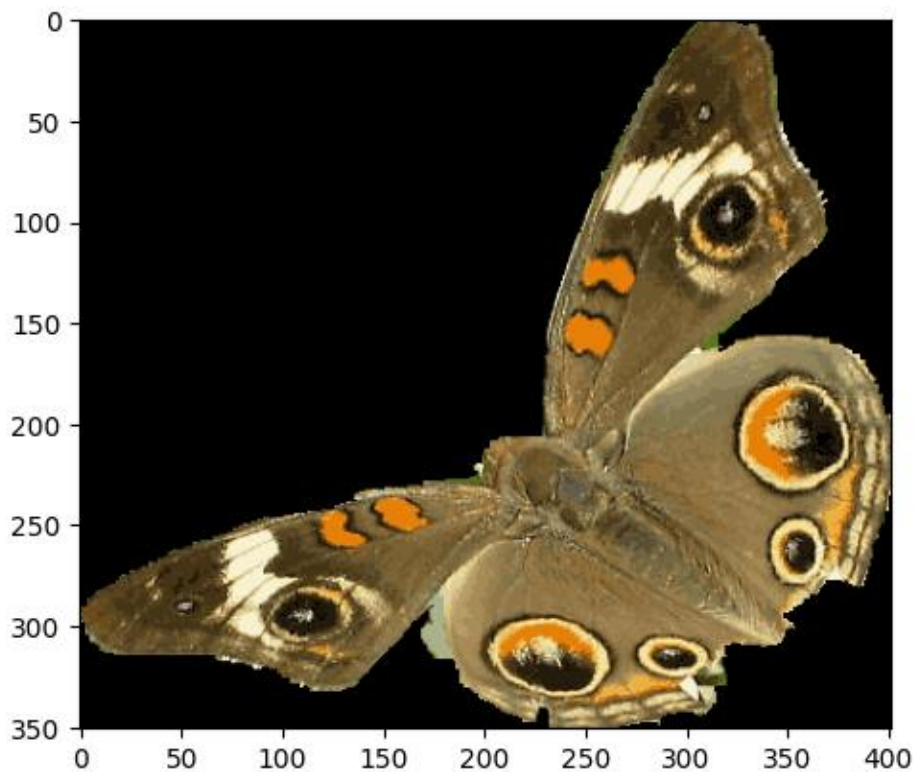


BÀI THỰC HÀNH SỐ 5

1. Sử dụng Kmeans Clustering để thực hiện Quantization các màu sắc cho bộ dữ liệu trên (Có thể thay đổi từ 8-16-64-256 màu).
2. Trích xuất đặc trưng, sau đó sử dụng KNN hoặc SVM để phân loại.
 - Thực hiện cắt ảnh với segmentation cho trước:



- Quantization màu sắc với số lượng 32 màu:



- Trích xuất đặc trưng trên tập dữ liệu và lưu vào file Data.csv:

```
df = pd.read_csv('/content/gdrive/MyDrive/Colab Notebooks/Data.csv')
df.head()
```

	0	1	2	3	4	5	6	7	8	9	...	246	247	248	249	250	251	252	253	254	labels	
0	0	0	0	0	0	0	2929	5094	0	0	...	0	0	0	0	0	0	0	0	0	0	001
1	0	0	0	0	0	0	0	2366	4789	0	...	0	0	0	0	0	0	0	0	0	0	001
2	0	1273	2086	585	0	1009	0	0	5529	1240	...	0	0	0	0	0	0	0	0	0	0	001
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	001
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	001

5 rows x 256 columns

- Chia dữ liệu thành training set và test set với tỷ lệ 80:20:

```
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2, random_state=1989)

sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

- Sử dụng KNN và SVC để phân loại:

```
1s classifier = SVC
Classifier = classifier(**params[classifier.__name__])
Classifier.fit(X_train, y_train)
y_pred = Classifier.predict(X_test)
print(classifier.__name__)
print(accuracy_score(y_test, y_pred))

SVC
0.3592814371257485

[25] 0s classifier = KNeighborsClassifier
Classifier = classifier(**params[classifier.__name__])
Classifier.fit(X_train, y_train)
y_pred = Classifier.predict(X_test)
print(classifier.__name__)
print(accuracy_score(y_test, y_pred))

KNeighborsClassifier
0.6227544910179641
```

- Kết quả không được tốt do dữ liệu từ trích xuất đặc trưng không được tốt.